(12) EUROPEAN PATENT APPLICATION

(71) Applicant: **International Business Machines
Corporation
Old Orchard Road
Armonk, N.Y. 10504(US)**

(72) Inventor: **Mathis, Joseph Richard**
725 Shady Hollow Drive
Georgetown, TX 78628(US)
Inventor: **Rouse, Gerald Lavelle**
3803 Curtis Drive
Round Rock, TX 78681(US)

(74) Representative: **Atchley, Martin John
Waldegrave
IBM United Kingdom Limited Intellectual
Property Department Hursley Park
Winchester Hampshire SO21 2JN(GB)**

(54) Two directional data communications system.

(57) The present invention relates to a two-directional data communications system comprising a first device (12) for transmitting and receiving data, a second device (14) for transmitting and receiving data, and communications links (16, 18) coupled to the first and second devices for transferring data therebetween.

According to the invention the communications system is characterised in that the first device comprises means for establishing a communications link over the communications system and for subsequently transferring data to the second device over the link and the second device comprises means for transferring, using the same communications link, a data transfer direction turnaround message to the first device and means for subsequently transferring data to the first device.
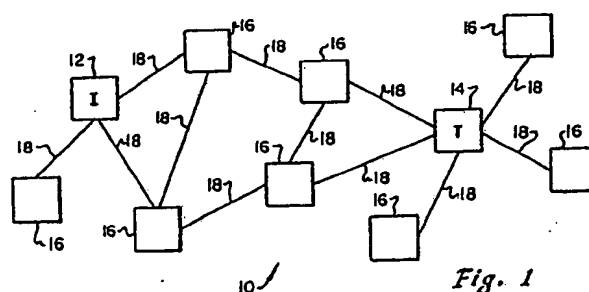
Fig. 1

TWO DIRECTIONAL DATA COMMUNICATIONS SYSTEM

The present invention relates to a data communications system for transferring data in two directions between nodes in the system, and more particularly to a communications system for transferring data between computer systems.

In communications systems for transferring data between computer systems, the communications links provided by the communications system can be represented as a graph with each attached computer system and specialized server device represented as a node. Data is transferred between the nodes using an agreed protocol which is determined by the designer of the system. Typically, data is transferred using synchronous serial communications. Data and control messages are bundled into discrete packets also called frames, each of which contains addressing information, data or a control message, and error checking values.

Many communications systems are dynamically switched, so that a connection path between two nodes on the system may pass through one or more intermediate switches. Different communication sessions between the same two nodes may be routed through different physical connection paths.

Because of various delays and competition for resources inherent in communications systems, various conflicts can arise. These conflicts involve the timing of data accesses to remote nodes, and can be important when certain shared resources, such as network file servers, are accessed.

Two-way transfer of data between nodes is generally accomplished as two separate transactions. When a local node requires to transfer information to a remote node, a communications link is set up for that transfer. Once the transfer is complete, the communications link is disconnected. If the remote node needs to transfer a reply, a new communications link is initiated and the transfer of the reply to the local node is made over the new link.

In some situations, it is necessary for a local node to be able to transmit data to and receive data from a remote node in a single transaction. This prevents interference from third party nodes during the time between closing of a transmission link in one direction and initiating a new link in the other direction. This situation is similar to certain bus conflicts found in computer systems, in which certain "atomic operations" must not be interrupted by other devices attached to the computer system bus. The operation "test-and-set" on multiprocessing systems typifies this type of atomic operation.

The object of the present invention is to provide an improved data communications system in which a single communications link can be used

for data transfer in both directions, a so called "atomic link" operation.

The present invention relates to a two-directional data communications system comprising a first device for transmitting and receiving data, a second device for transmitting and receiving data, and communications links coupled to the first and second devices for transferring data therebetween.

According to the invention the communications system is characterised in that the first device comprises means for establishing a communications link over the communications system and for subsequently transferring data to the second device over the link and the second device comprises means for transferring, using the same communications link, a data transfer direction turnaround message to the first device and means for subsequently transferring data to the first device.

In order that the invention may be more readily understood an embodiment will now be described with reference to the accompanying drawings, in which:

FIG. 1 is a block diagram of a communications system,

FIG. 2 is a flowchart illustrating the operation of a data transfer direction turnaround on a communications link between nodes in the system illustrated in Fig. 1, and

FIG. 3 is a diagram illustrating the timing and control signals of a communication session between the two nodes, referred to in Fig. 2.

Referring to Figure 1, a data communications network or system 10 contains a plurality of nodes. An initiating node 12 initiates a communications session with a target node 14 as described below. The remaining intermediate nodes 16 form the rest of the network 10.

Each of the nodes 12, 14, 16 may be a computer system, a terminal, a printer, a file server, or any other type of device which may be connected together to form a network. A node 16 may also represent a gateway to another network.

The nodes are interconnected by wired connections 18. The network shown in Figure 1 is not intended to be representative of any particular network arrangement. The arrangement to be described is equally applicable to star, token-ring, packet switched, and other network topologies. The network 10 is preferably dynamically switched, so that two different communication sessions for transferring data between nodes 12 and 14 may use different sets of connections 18 and intermediate nodes 16.

In the description which follows, initiating node 12 will also be referred to as node I. Target node

3
  4

14 will also be referred to as node T.

  The communication protocol described in connection with Figures 2 and 3 is described in detail in copending European Patent Application No. , (reference AT9-88-067), which is incorporated by reference herein. The protocol described in the copending application is utilized herein as a preferred embodiment, but is not required to practice the present invention.

  In the communications protocol to be described, a communications link is established by a sending device which wishes to transfer data to a receiving device. The sending device initiates the transfer by sending a control message to the receiving device, which responds by returning a control message containing parameters defining the environment of the receiving device. This environment can include information such as the number of receive buffers available in the receiving device, and their size.

  Once the sending device has received these environment parameters, it sends data in blocks which can be accommodated in the receiver's buffers. As the receiver removes each block of data from a buffer, it transmits an acknowledge message to the sender. By keeping track of the acknowledge messages, the sender can control the data transfer and detect certain types of transfer errors. The details of such communications protocol are described further in the above referenced copending application.

  Referring to Figure 2, a flowchart illustrating data transfer by an "atomic link" operation between the initiating node 12 and the target node 14 is shown. Node I requests the atomic link operation, and node T permits it and performs the data transmission direction turnaround. At step 26 node I initiates a communication with node T by requesting a communications link. This is accomplished by sending a control message to node T requesting communication. At step 28 node T determines whether it will accept a link request. If not, at step 30 node T denies the request and at step 32 disconnects the link. If node T will accept the request, at step 34 node T sends parameters which describe its operating environment. These parameters preferably include the number of available buffers in node T, and their size.

  Once it has received the parameters, at step 36 node I sends its request to node T. This request is sent as data which is transferred to the target node T in the normal manner for data transfers. The request asks the target node T to perform an atomic link operation consisting of changing the direction of data communication between the initiating and target nodes without terminating the communication link and establishing a new link when the transfer of data from node I to node T has been completed..

  At step 38, the target node T decides whether it will accept the atomic link operation request. This decision can be based on a number of factors, such as whether the target node T is capable of supporting such a request at the current time, and whether the initiating node I has the authority to request such an operation. If, for whatever reason, the target node T does not accept the request, at step 40 it sends a control message to node I denying the request and at step 42 it disconnects the communications link.

  If the target node T will accept the request, at step 44 it transmits a turnaround message to the initiating node I. The turnaround message provides several functions. It acts as an acknowledgement of receipt of the last data block sent by the initiating node I. It also indicates that the request for an atomic link operation was granted, implying that data transfer will now change direction, and acts as a request to transfer data to the initiating node I.

  In response to the turnaround message, at step 46 the initiating node I transmits its receiving parameters to the target node T. These parameters preferably include the number of available buffers in node I, and the buffer size. At step 48 the target node T then sends data to the initiating node I. The final block of data contains a flag which indicates that it is the final data block of this transfer. When the initiating node I acknowledges the final block of data, at step 50 it disconnects the communication link and the atomic link operation is complete.

  Figure 3 shows the timing and preferred data and control messages used in a data transfer operation such as described in connection with Figure 2. The message RQ_SEND is a control message requesting permission to transmit information. RS_SEND is a control message granting permission to transfer data, and providing the environment parameters described above. The message SEND is a block of data, also referred to as a data frame, being transferred. The signal RS_ACK is an acknowledgement message indicating that a data frame has been received and emptied from the receive buffer.

  RS_ACK_T is the turnaround message described in connection with Figure 2. The SEND-(data end) message is a data frame which includes a flag indicating that this is the final data frame of a transfer operation. The RS_ACK(disconnect) message is an acknowledgement signal sent in response to receipt of a SEND(data end) data frame message, and acts to disconnect the communications link between the initiating node I and the target node T.

  The circles 60 and 62 indicate interrupts in the operation of the initiating node I, and the squares 64, 66, 68, and 70 indicate interrupts in the opera-

5

10

15

20

25

30

35

40

45

50

55

tion of the target node T. These interrupts 60 through 70 indicate times at which it may be necessary for the processor controlling the corresponding node to perform a software routine prior to continuing the communication process. This routine typically makes decisions about the process, including deciding whether to continue.

The initiating node I begins the data transfer by sending an RQ_SEND message to the target node T. This establishes the communications link and requests permission to transfer data to the target node T. Interrupt 64 is raised at the target node T to allow its processor to determine whether to accept a communication. If the target node T does not grant such permission, it sends a reply message RS_NAK (not shown). If the target node T grants permission to send data, it indicates this fact and defines its environment using an RS_SEND message. The initiating node I then begins sending data with a series of SEND messages. In Figure 3, the sending of only a single SEND frame message is shown, but any number of data frames may actually be transmitted by the initiating node I at this time. If more than one SEND frame is sent to the target node T, the initiating node I expects to receive an RS_ACK message for each frame sent, except for the last one. The last frame sent contains a flag indicating that it is the last data frame.

Included in the last frame of data sent from the initiating node I to the target node T is a request to perform an atomic link operation. When the SEND-(data end) message is received by the target node T, interrupt 66 causes the processor associated with the target node T to analyze the request for an atomic link operation included in the data frame sent. If the target node T does not accept the request, a control message RS_NAK (not shown) is returned to the initiating node I, indicating that the request is not acknowledged. The RS_NAK message also causes the communications link to be disconnected. In the example shown in Figure 3, the request is acknowledged, and the control message RS_ACK_T is sent to the initiating node I. This message causes an interrupt 60 in the operation of the initiating node, which signals that the previous SEND sequence was successful and the request for an atomic link operation is granted, and allows the initiating node I to undertake whatever processing is necessary prior to receiving data.

As described above, one of the functions of the RS_ACK_T message is to initiate a data transfer from the target node T to the initiating node I. After the interrupt 60, the initiating node I sends an RS_SEND message, which defines its operating environment and gives the target node T permission to begin sending data. Interrupt 68 is used by the target node T to process this grant of permis-

sion, after which the target node T begins sending data in SEND frames.

The initiating node I responds to the receipt of each SEND frame message with an RS_ACK message, which is used to verify correct data transfer as described above. The last data block is sent with a SEND(data end) message, which is acknowledged by an RS_ACK(disconnect) message. The disconnect signal raises interrupts 62, 70 to both nodes I, T to indicate that data transfer is complete.

In reply to the SEND(data end) message sent by the target node T, it is possible for the initiating node I to reply with an RS_ACK_T message instead of an RS_ACK(disconnect) message. This message would request another reversal of direction of data transfer, allowing the initiating node I to send further data to the target node T. The process of sending consecutive blocks of data, followed by an RS_ACK_T message, may be repeated as often as desired. This would allow a large number of data transfer direction reversals to be made without disconnecting the current communications link and establishing a new one. However, in most cases it will not be necessary to perform more than a single turnaround. In addition, an extended operation in which the communications link is locked may have undesired effects upon operation of the remainder of the network.

As will be appreciated by those skilled in the art, the arrangement described above provides for an operation on a data communication network which locks a communications link for bidirectional data transfer. A special control command is used to turn the data transfer direction around. Such an operation can be used when it is necessary to transfer data in both directions between two nodes on a network without the possibility of an intervention by a third node. This operation is similar to a test-and-set command used in a multiprocessing computer system, except that it operates over a distributed network. In a communications system which requires each data packet to be separately acknowledged, a modified acknowledge message can be used to signal the data direction turnaround.

Claims

1. A two-directional data communications system, comprising
a first device (12) for transmitting and receiving data,
a second device (14) for transmitting and receiving data, and
communications links (16, 18) coupled to said first and second devices for transferring data therebetween,
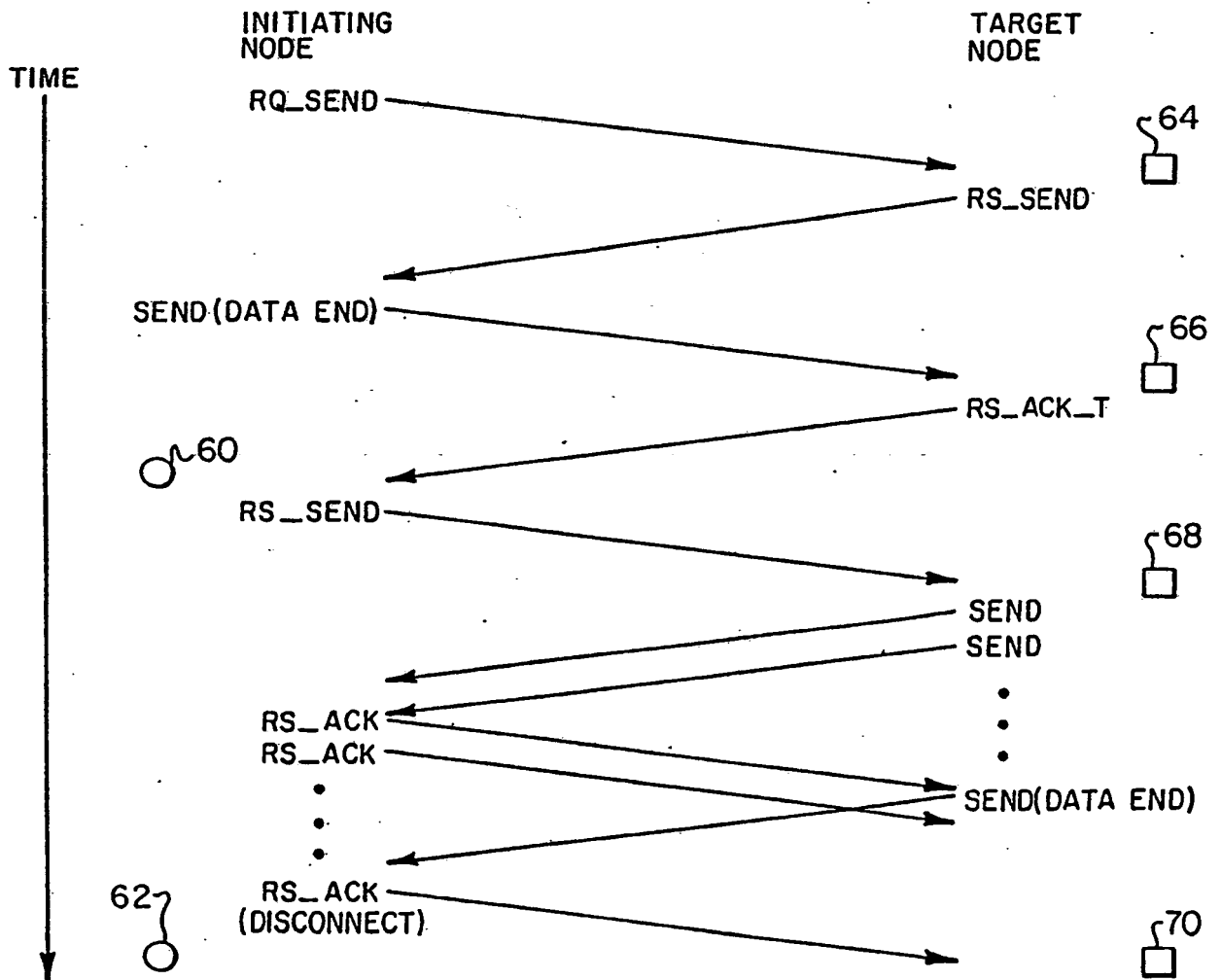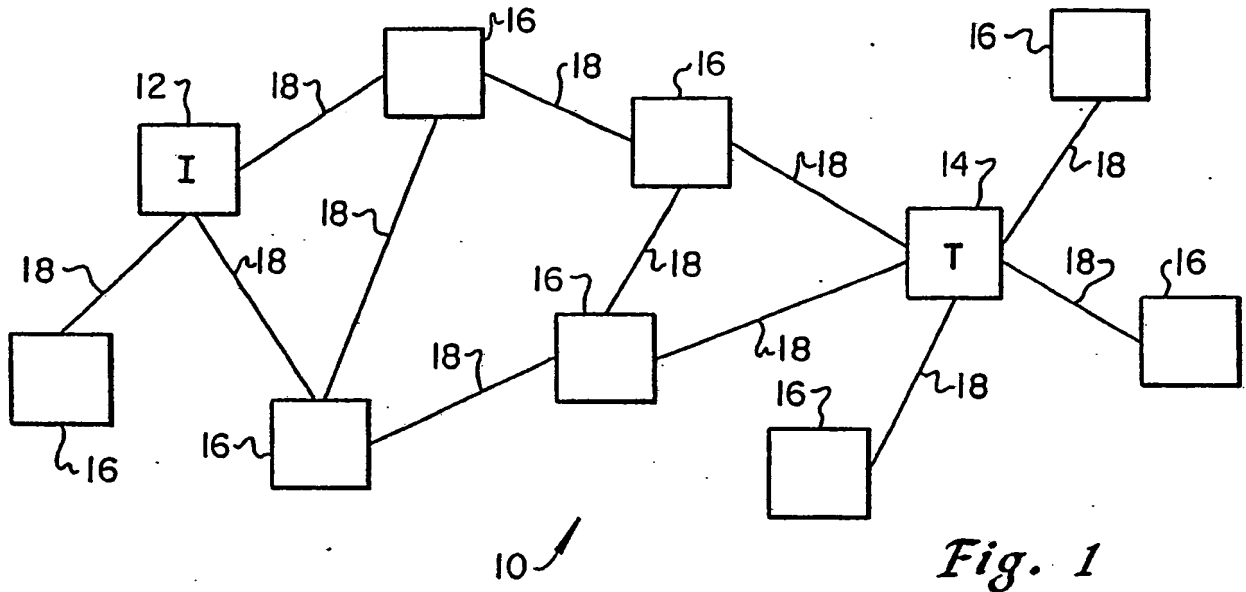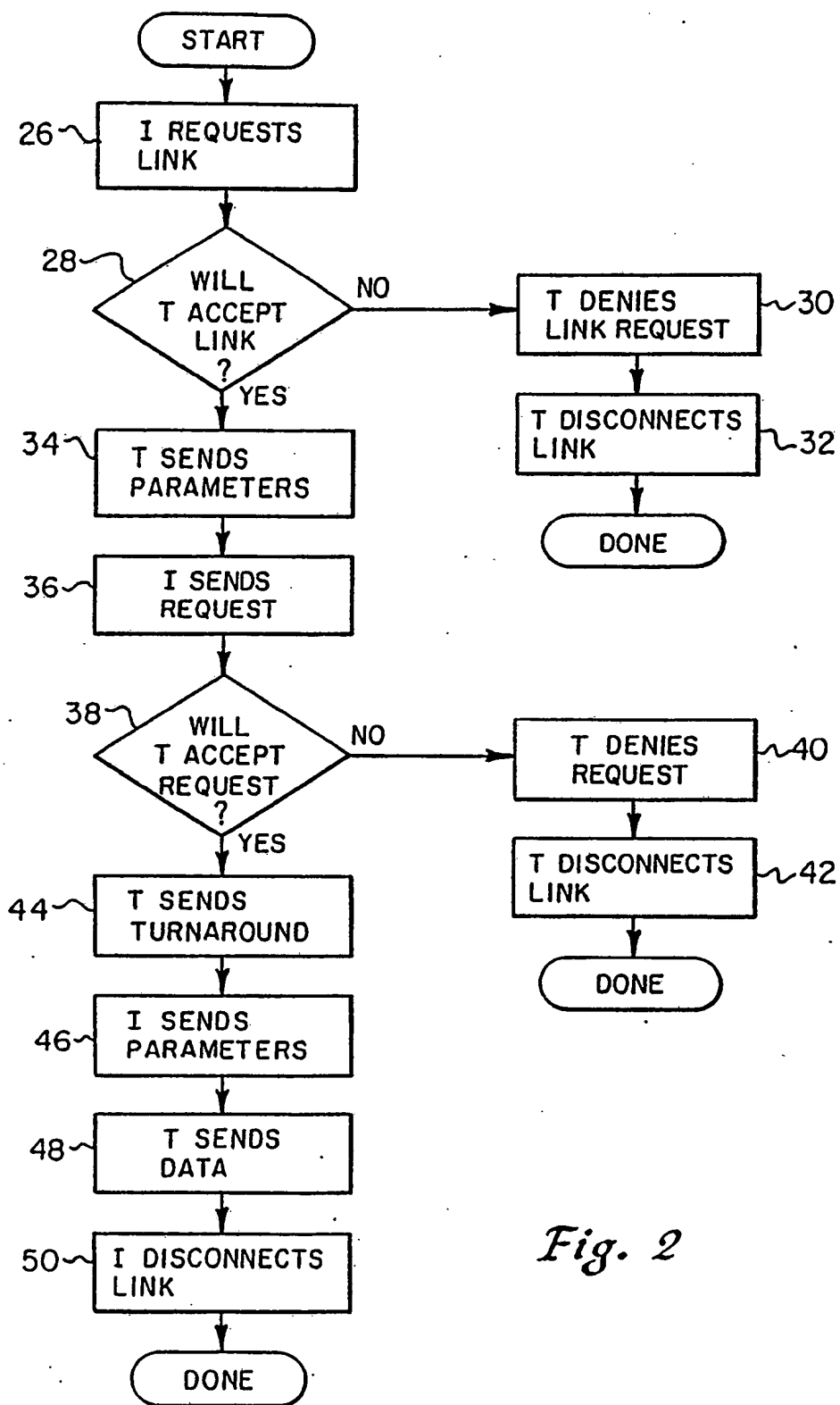
characterised in that
said first device comprises means for establishing
a communications link over said communications
system and for subsequently transferring data to
said second device over said link, and
said second device comprises means for transfer-
ring, using the same communications link, a data
transfer direction (14) turnaround message to said
first device (12) and means for subsequently trans-
ferring data to said first device.

2. A data communications system as claimed
in Claim 1 characterised in that said means for
transferring said data transfer direction turnaround
message is responsive to the receipt from said first
device (12) of a message indicating that the trans-
fer of data from said first device has been com-
pleted.

3. A data communications system as claimed
in either of the preceding claims characterised in
that said first device (12) comprises means for
transferring to said second device (14) a message
requesting that operation of the change of data
transfer direction should take place when the trans-
fer of data from said first device to said second
device has been completed.

Fig. 1

Fig. 3

START

26 — I REQUESTS LINK

28 — WILL T ACCEPT LINK ? —— NO ——▶ T DENIES LINK REQUEST ~30

T DISCONNECTS LINK ~32

DONE

YES

34 — T SENDS PARAMETERS

36 — I SENDS REQUEST

38 — WILL T ACCEPT REQUEST ? —— NO ——▶ T DENIES REQUEST ~40

T DISCONNECTS LINK ~42

DONE

YES

44 — T SENDS TURNAROUND

46 — I SENDS PARAMETERS

48 — T SENDS DATA

50 — I DISCONNECTS LINK

DONE

Fig. 2